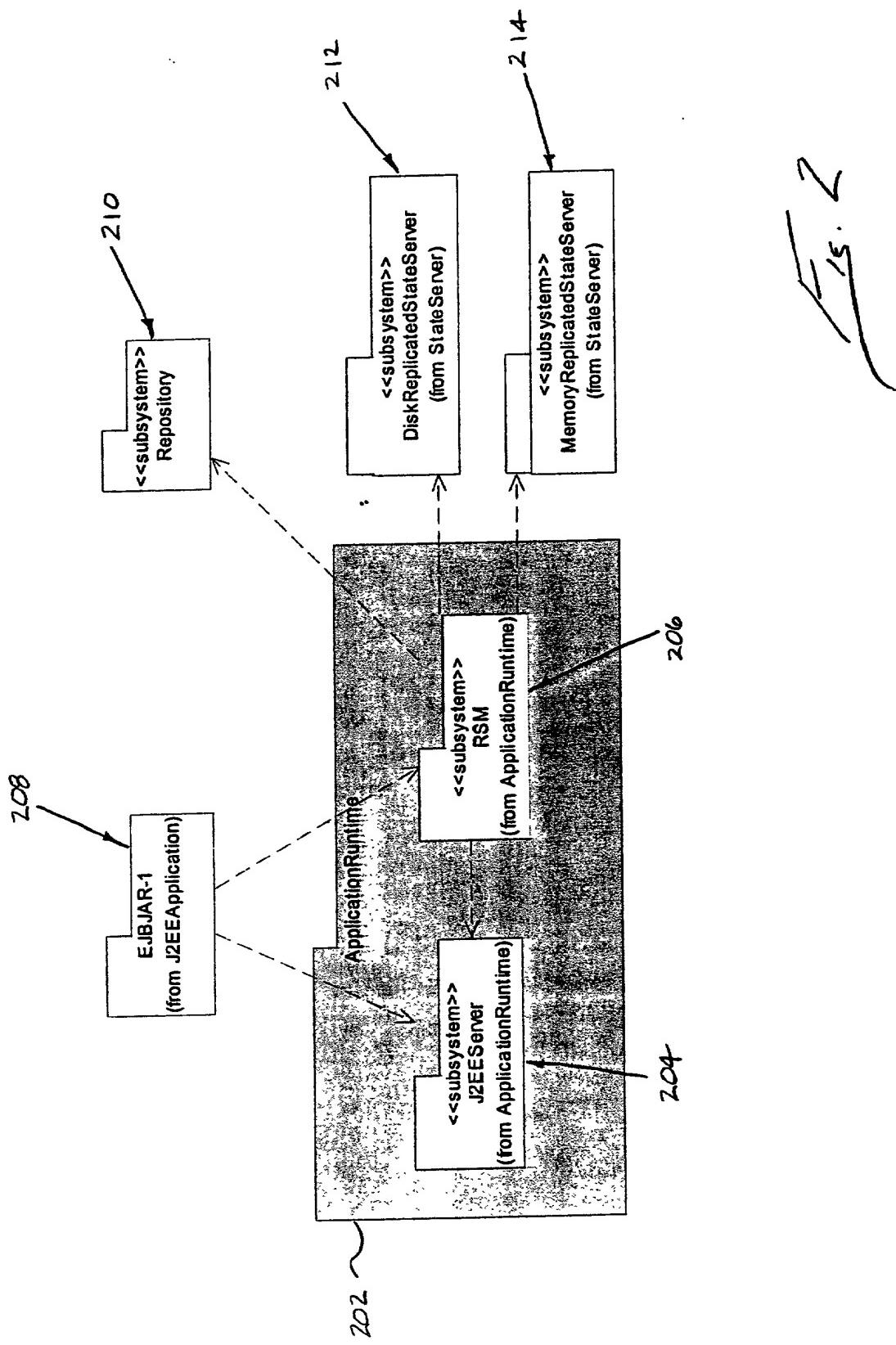
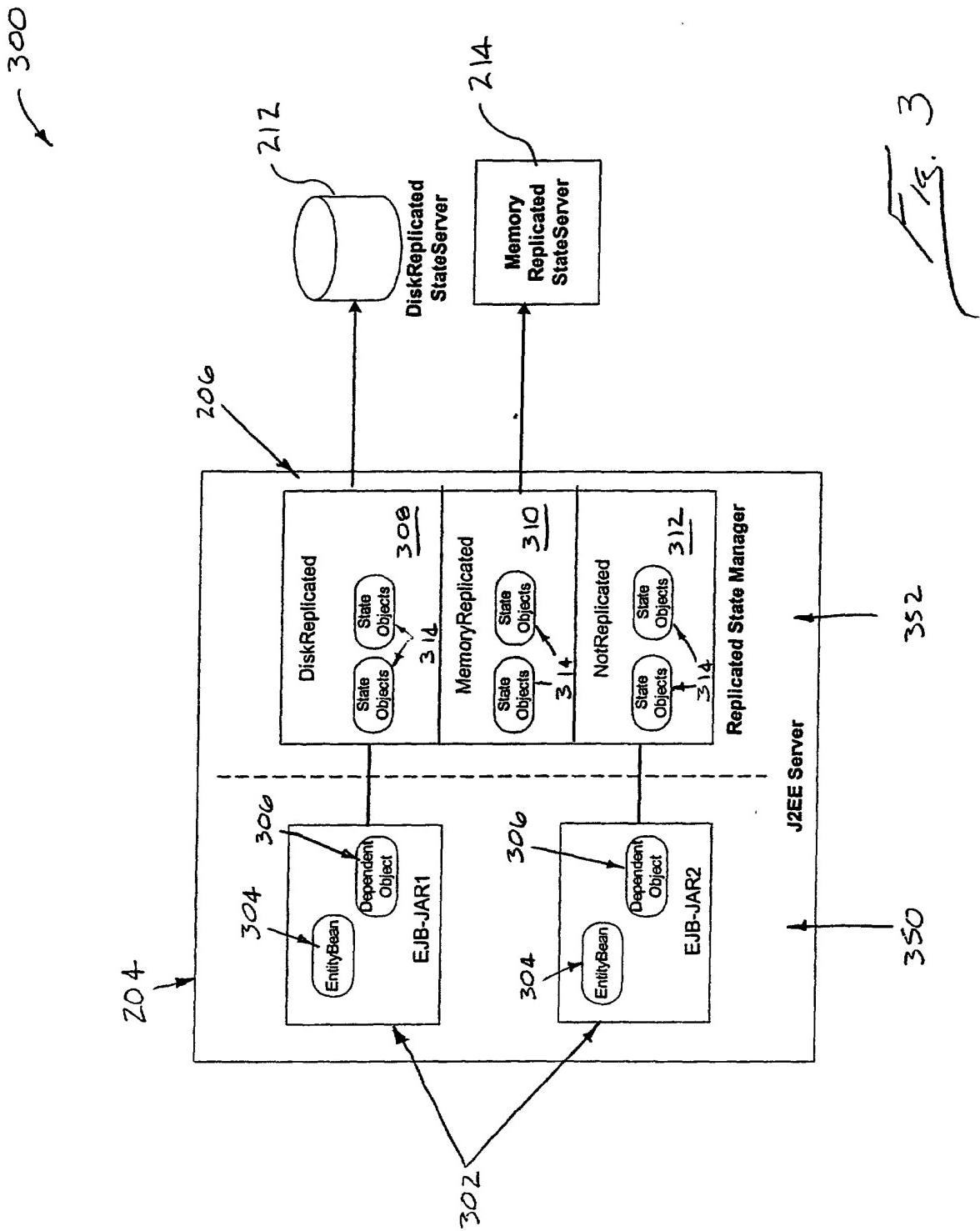


FIGURE 1
(Prior Art)

200





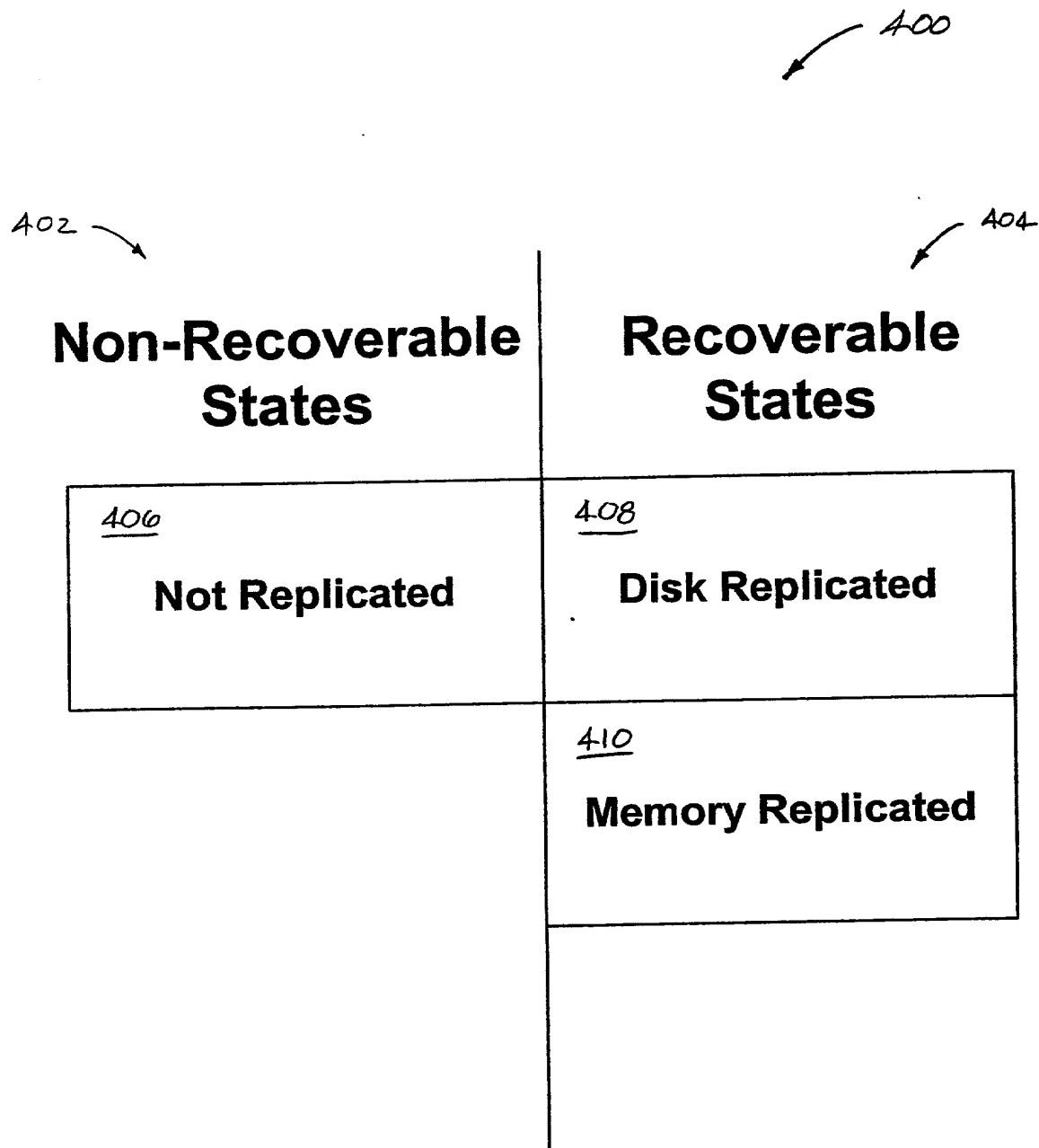
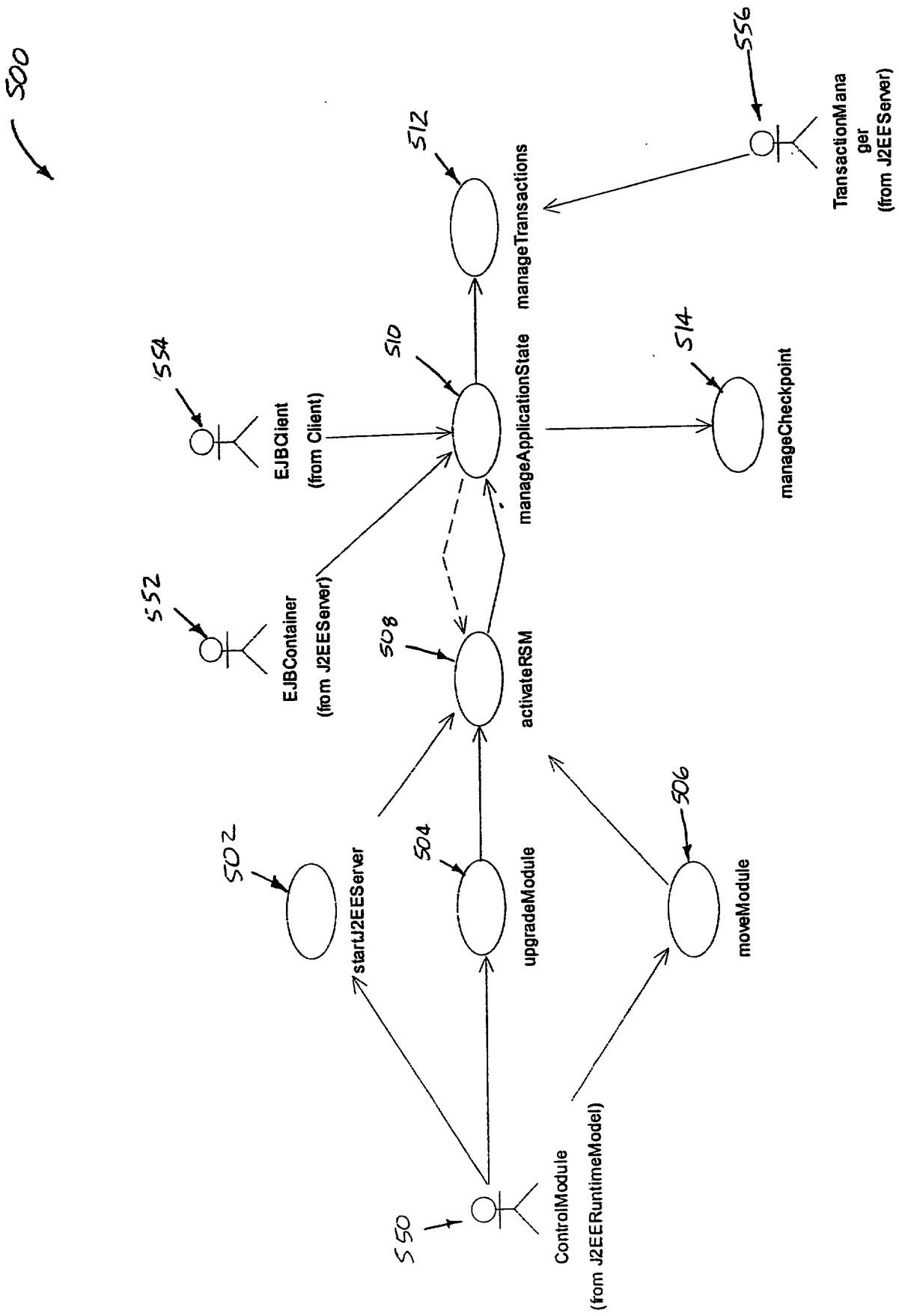


FIG. 4

5



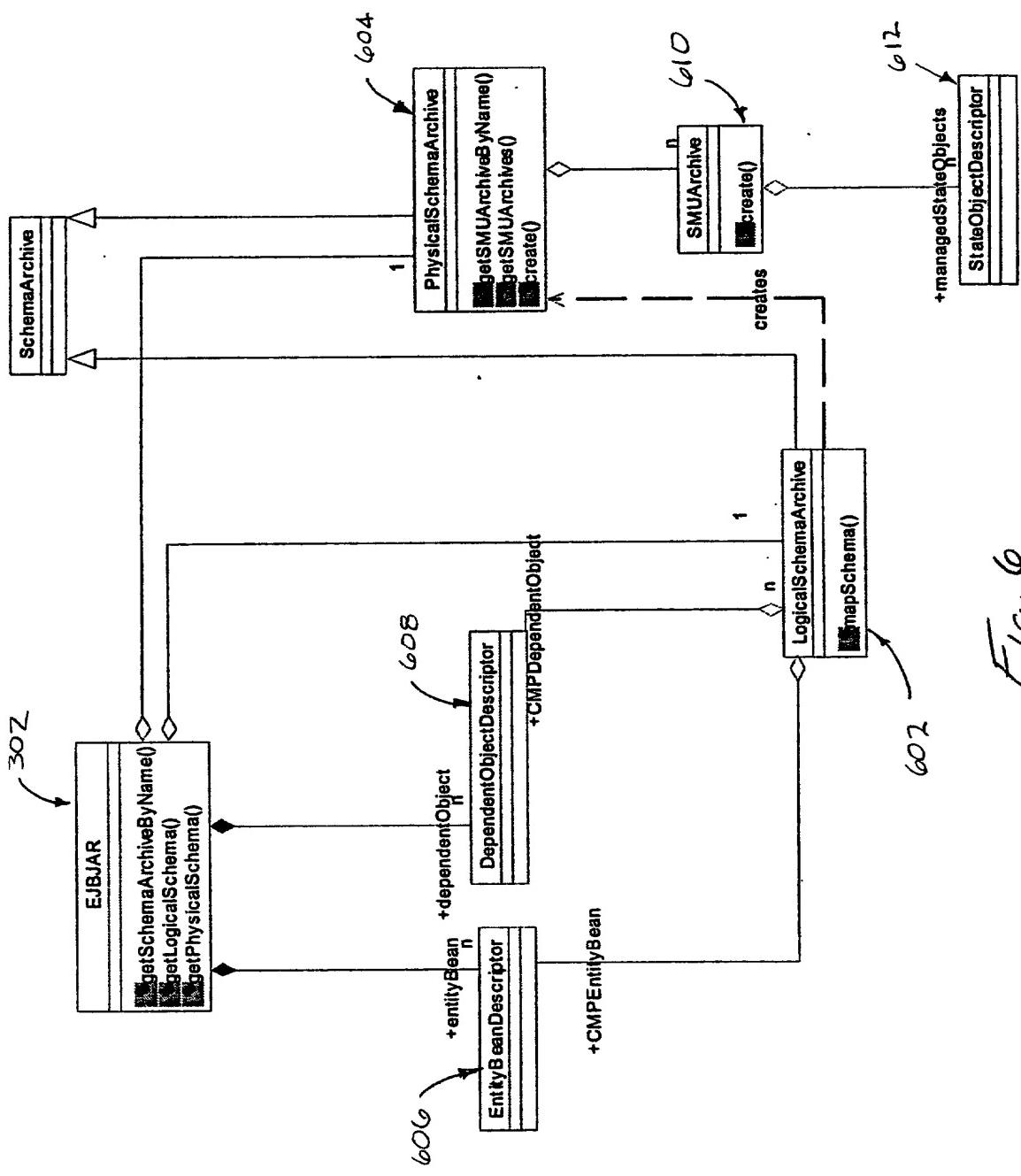


Fig. 6

700

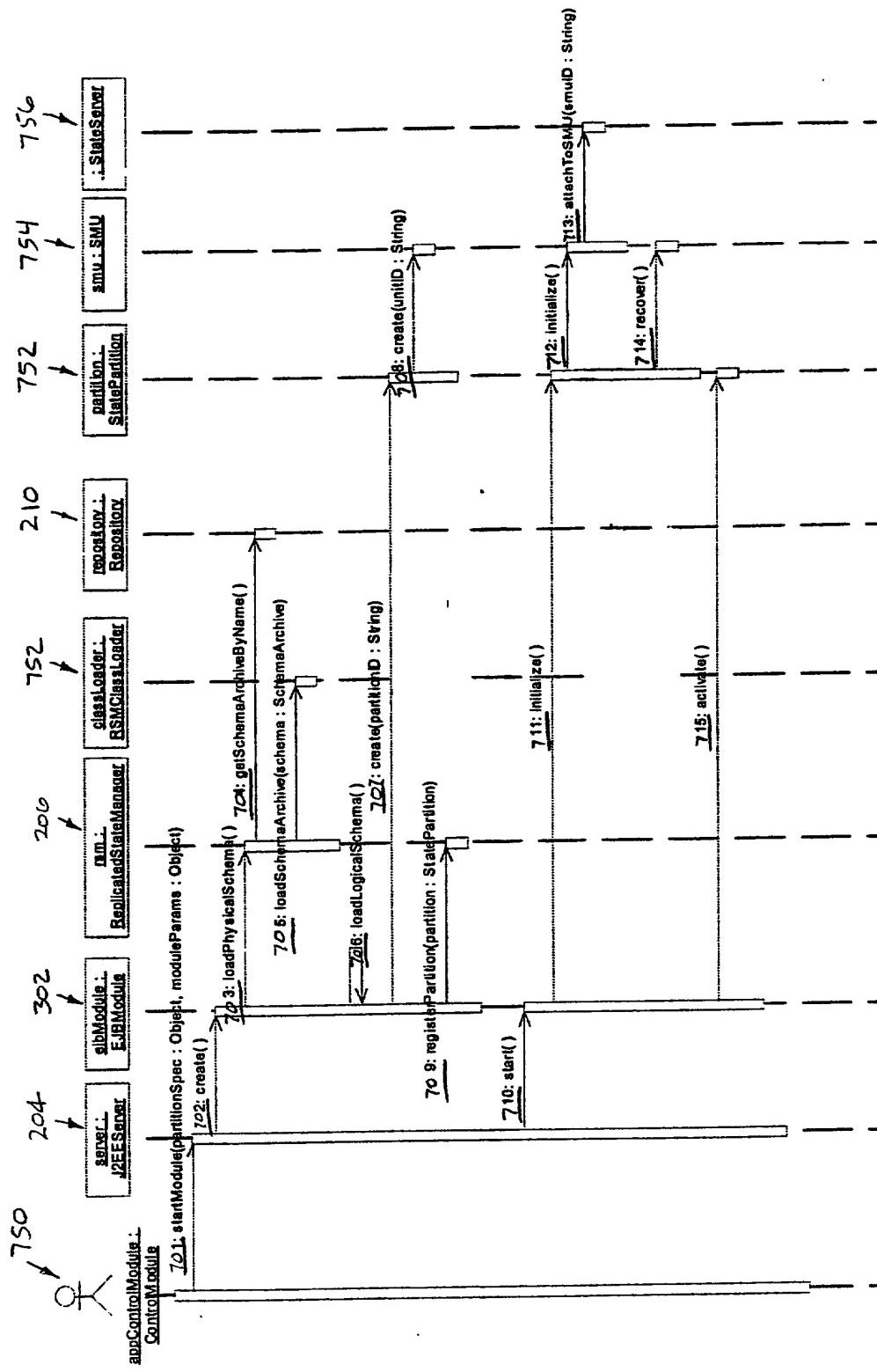


Fig 7

800

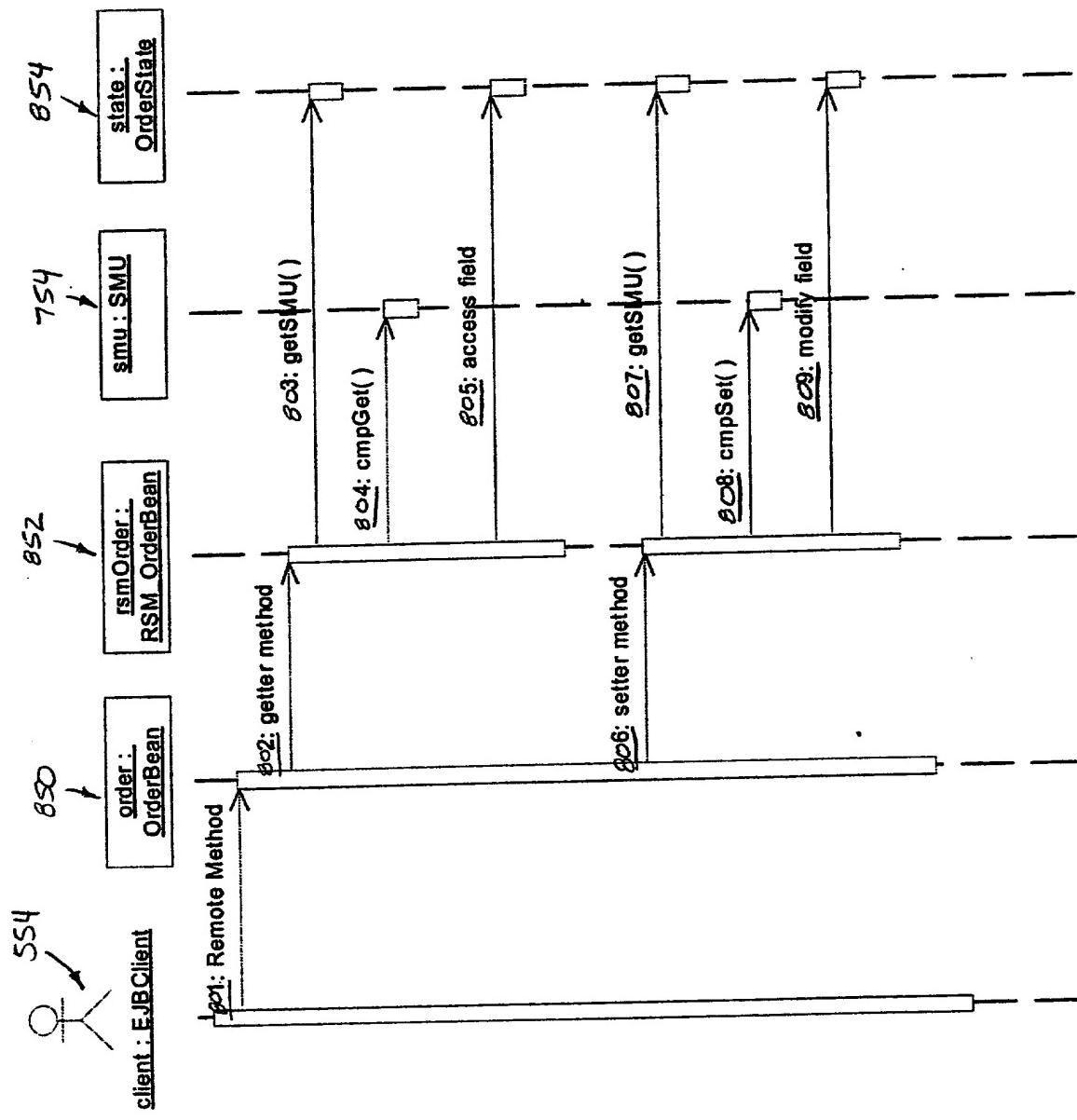


Fig. 8

900

Fig 9

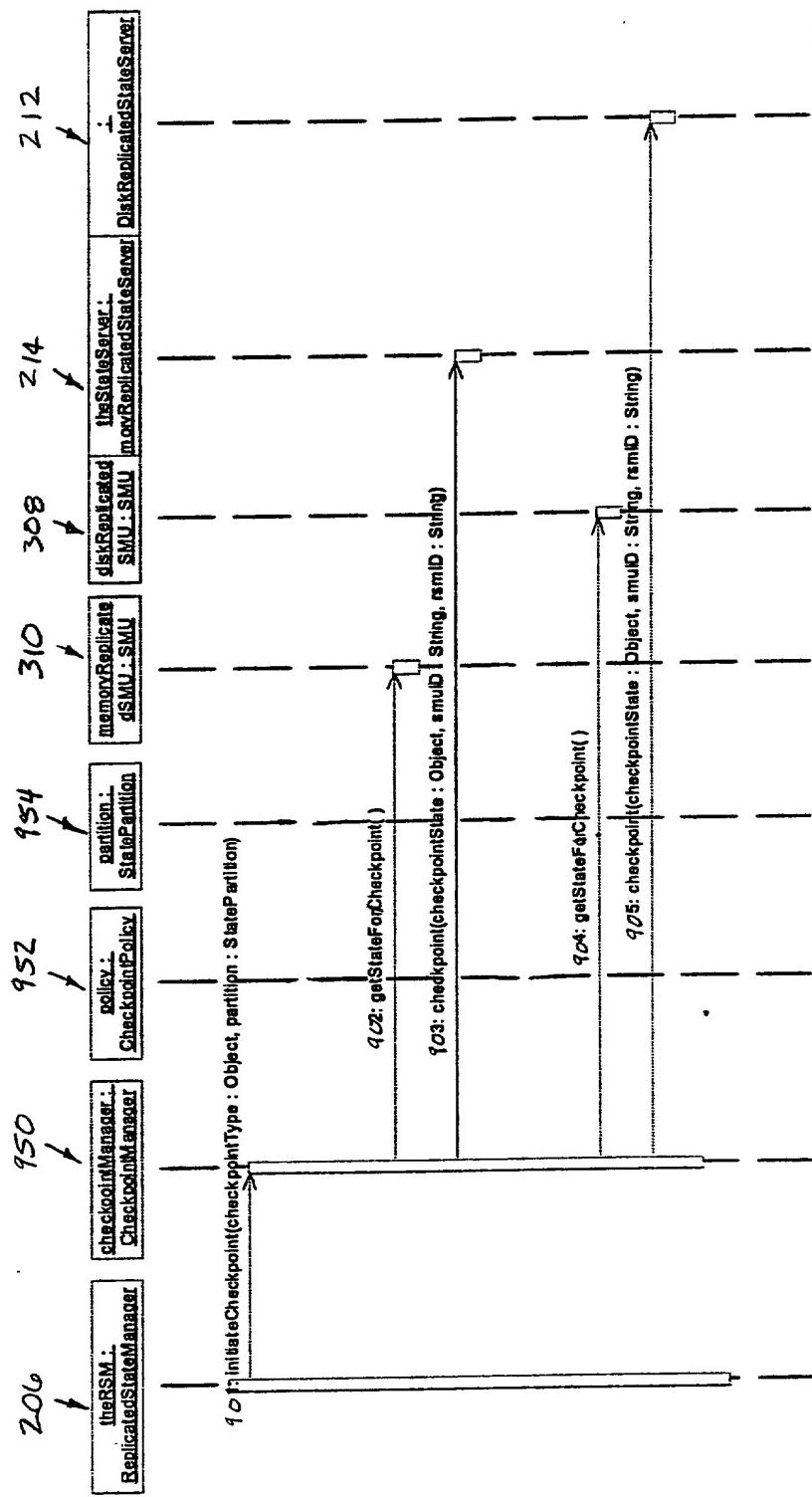


图 12.10

```

sequenceDiagram
    actor EJBClient
    actor EJBContainer
    actor Transaction
    actor RSMOrderbean
    actor Synchronization
    actor FSM
    actor StatePartition

    EJBClient->>EJBContainer: Remote Method
    activate EJBContainer
    EJBContainer->>Transaction: begin()
    activate Transaction
    Transaction->>RSMOrderbean: lock()
    activate RSMOrderbean
    RSMOrderbean->>Synchronization: transactionStarted(tx : Transaction)
    activate Synchronization
    Synchronization->>FSM: registerSynchronization(sync : Transaction)
    activate FSM
    Synchronization->>StatePartition: lockPartition(statePartition : StatePartition)
    activate StatePartition
    StatePartition->>FSM: lock()
    activate FSM
    FSM->>RSMOrderbean: usePartition(listOfPartitions : Object)
    activate RSMOrderbean
    RSMOrderbean->>FSM: dispatch()
    activate FSM
    RSMOrderbean->>Synchronization: Method dispatch
    activate Synchronization
    Synchronization->>FSM: beforeCompletion()
    activate FSM
    Synchronization->>StatePartition: unlockPartition(statePartition : StatePartition)
    activate StatePartition
    StatePartition->>FSM: unlock()
    activate FSM
    FSM->>Synchronization: transactionConfirmed(tx : Transaction)
    activate Synchronization
    Synchronization->>EJBContainer: commit()
    deactivate EJBContainer
    deactivate Transaction
    deactivate RSMOrderbean
    deactivate Synchronization
    deactivate FSM
    deactivate StatePartition

```

The sequence diagram illustrates the execution flow between several components:

- EJBClient** initiates a **Remote Method** call to the **EJBContainer**.
- EJBContainer** begins a **begin()** transaction.
- EJBContainer** locks the **RSM Orderbean**.
- RSM Orderbean** starts a **transactionStarted(tx : Transaction)** synchronization.
- RSM Orderbean** registers the **Transaction** as a synchronization point.
- RSM Orderbean** locks the **StatePartition**.
- RSM Orderbean** uses the partitioned object.
- RSM Orderbean** dispatches the method.
- RSM Orderbean** sends a **Method dispatch** message to the **Synchronization**.
- Synchronization** sends a **beforeCompletion()** message to the **FSM**.
- Synchronization** unlocks the **StatePartition**.
- FSM** confirms the transaction.
- FSM** sends a **commit()** message back to the **EJBContainer**.
- The **EJBContainer** completes the transaction.

图 12.10